GAMs (Generalized Additive Models)

STAT 245

Motivation

- We can model continuous, logical, and count response variables
- We can include quantitative and categorical predictors
- What about nonlinear relationships?

Already nonlinear

Categorical predictor variables

- Making use of indicator variables for (all but one of the) categories, we can model a situation where each value of the predictor variable has a different effect on the response.
- But...if forcing a quantitative variable to be categorical...
  - How many categories?
  - What about periodicity?

Already nonlinear

GLMs

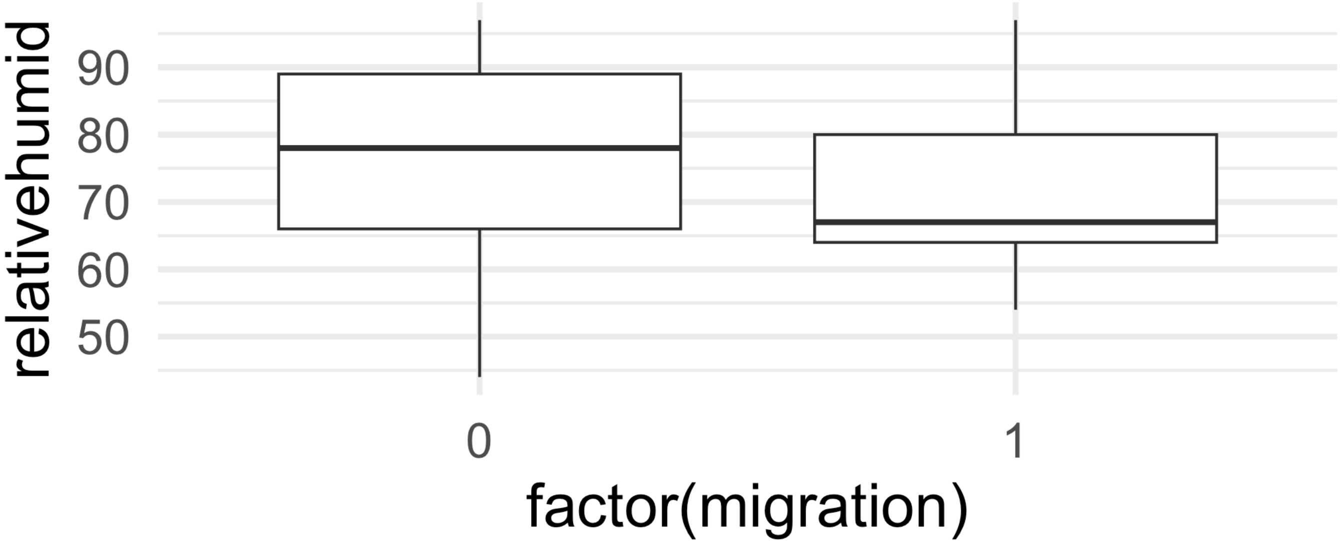
- In binary or count regression, predictor-response relationship is linear on the scale of the link function (= scale of the RHS of the equation)
- But non-linear on the scale of the response variable (LHS)
- Well - Nonlinear, but always **monotonic**

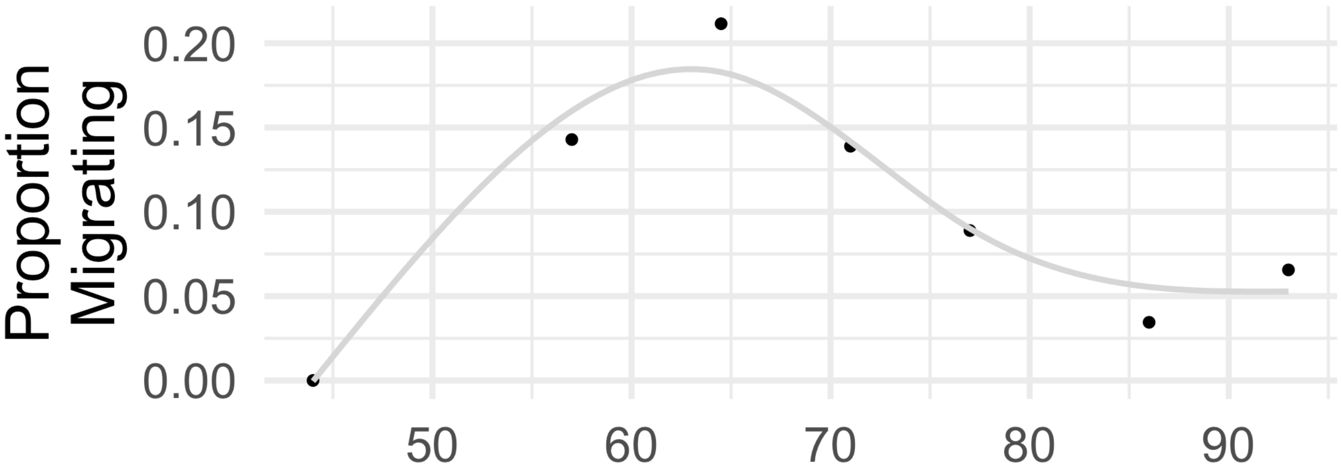
Not going there

Popular options that we won't pursue (and why)

- Transformations of predictors or response (log, powers, etc.)
- Polynomials (adding predictor^2, predictor^3, etc.)
- Good *with theoretical justification*, otherwise hard to choose
- We want one flexible solution to get any shape
- We want easily interpretable results

Non-linear, non-monotonic
Example: Bat migration & Weather





Our case study: GRR weather station + metadata

```
grweather <- read_csv('https://sldr.netlify.app/data/grr-weather-metric.csv',
         show_col_types = FALSE) |>
    select(SNOW, TMIN, DATE, TSUN) |>
  mutate(MONTH = lubridate::month(DATE),
         DAY = lubridate::yday(DATE),
         YEAR = lubridate::year(DATE),
         WEEK = lubridate::isoweek(DATE),
         PREV.SNOW = lag(SNOW)) |> na.omit()
```

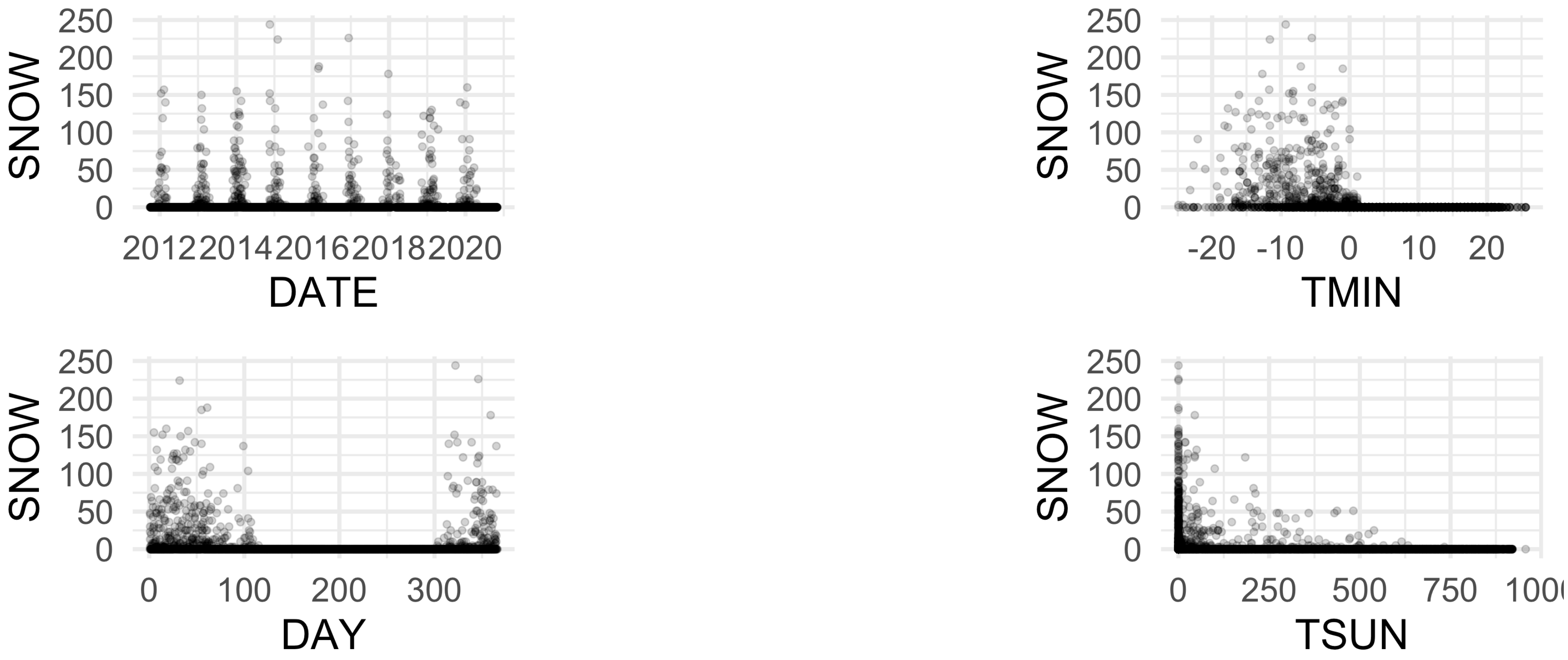
```
gf_point(SNOW ~ DATE, data = grweather, alpha = 0.2)
```

```
gf_point(SNOW ~ DAY, data = grweather, alpha = 0.2)
```

```
gf_point(SNOW ~ TMIN, data = grweather, alpha = 0.2)
```

```
gf_point(SNOW ~ TSUN, data = grweather, alpha = 0.2)
```

Snowy Weather Case Study



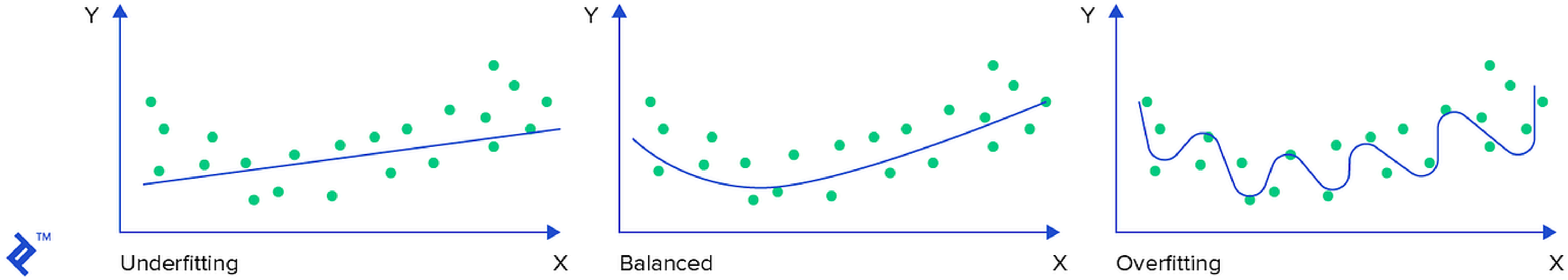
Smooth functions

- Goal: predictor-response relationship can have any shape
- linear, or nonlinear with *any* shape
- user control of "wiggliness"

Basis functions

- Several parts, or *basis functions*, sum to form a smooth
- Each has simple shape
- Scaled and added together, yield nearly any shape
- Higher basis dimension (more functions added together) can = more wiggles
- Goal: enough flexibility to fit data well, without *overfitting*

Overfitting



Fitting GAMs

Generalized Additive Models

- R Package `mgcv`, function `gam()`
- Can fit any family (linear, binary, count) depending on response variable type
- Smooth Types: App
- More background (workshop materials)

Join by Web **PollEv.com/stacyderuite335** Join by Text Send **stacyderuite335** to **37607**

Which terms should I plan as smooths in a regression model?

0

(A) Decide based on exploratory graphics

0%

(B) Decide based on background knowledge of the scenario

0%

(C) Just make them all smooths, SEE MORE ze allows

Model formula syntax

Function `s()` specifies a smooth. Inputs:

- variable name(s)
- `k` (see `?choose.k`)
- `bs`
- `by`

How do we choose? see App ; Defaults: `tp` or `cs` as default options, or `cc` for cyclic

Break my App

- What choices of smooth type and/or `k` work *well* for your variable?
- What choices of smooth type and `k` can result in a smooth that seems overfitted, or mismatched to data in some other way?
- What does shrinkage (`bs = _s`) do?

https://connect.cs.calvin.edu/DATA545/smooth/

`gam()` **Options**

We can also fit the model and smooths by different methods and with options:

- ~~`method = 'GCV.Cp'`~~
- ~~`method = 'REML'`~~
- `method = 'ML'`
- `select = TRUE` (or ~~`FALSE`~~)

GAM Example - Snow

```r
library(mgcv)
snow.gam <- gam(SNOW ~ s(DAY, k = 20, bs = 'cc') +
                s(TMIN, k = 5, bs = 'cs') +
                s(YEAR, k = 5, bs = 'cs') +
PREV.SNOW,
              data = grweather,
              method = 'ML',
              select = TRUE)
```

When poll is active respond at **PollEv.com/stacyderuite335** Send **stacyderuite335** to **37607**

Our model has only smooth terms. If we wanted, could we also include "regular" linear quantitative predictors? Could we include categorical predictors?

0

GAMs can have categorical predictors

0%

GAMs can have linear terms

0%

GAMs can have both categorical and linear terms as well as smooths; this example just happens not to have any.

0%

When poll is active respond at **PollEv.com/stacyderuite335** Send **stacyderuite335** to **37607**

We need about one basis dimension ("k") per "wiggle" in a smooth. What strategies help us choose k?

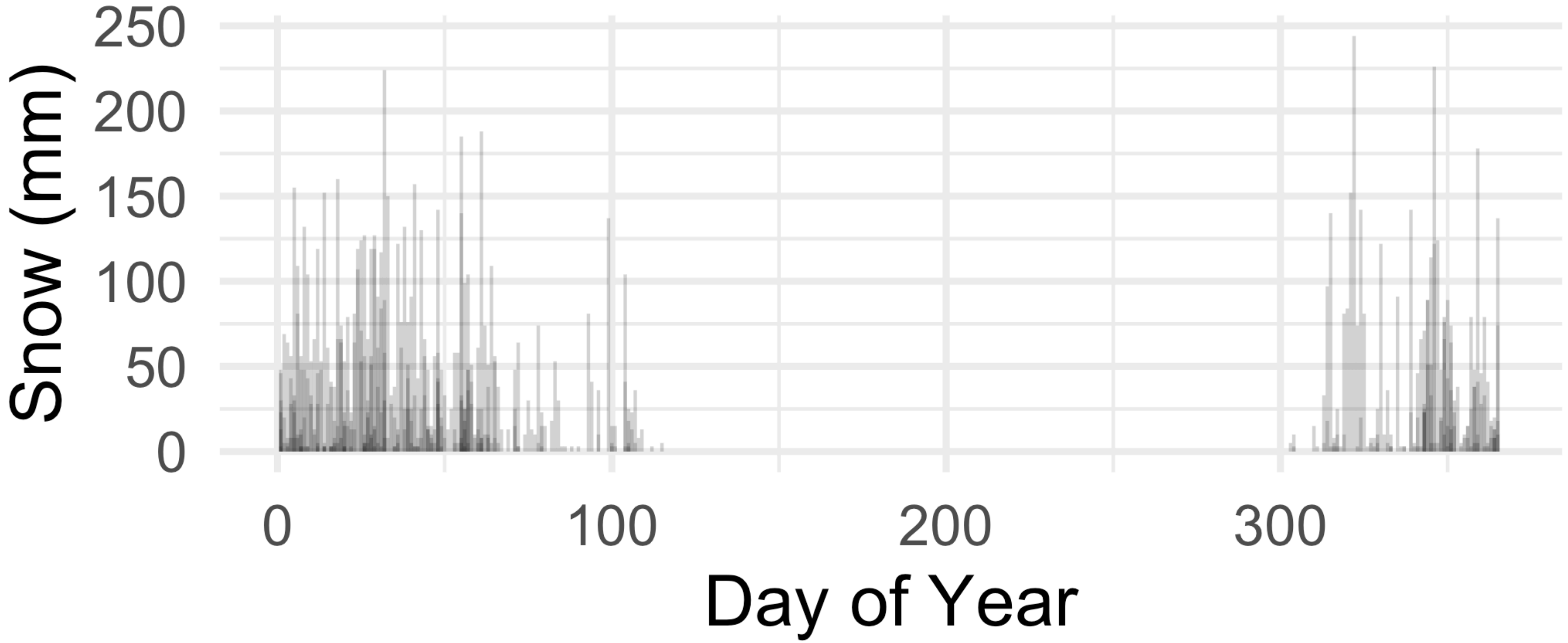
Consider size of dataset (about (k-1) coefs, (k-2) for cyclic)	0
Consider wiggliness of expected relationship (want silk, or carpet?)	0
Use k a little bigger than you think you need, but with select = TRUE and shrinkage	0

```r
summary(snow.gam)
```

```
## 
## Family: gaussian 
## Link function: identity 
## 
## Formula:
## SNOW ~ s(DAY, k = 20, bs = "cc") + s(TMIN, k = 5, bs = "cs") + 
##     s(YEAR, k = 5, bs = "cs") + PREV.SNOW
## 
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  3.66340    0.31782   11.53   <2e-16 ***
## PREV.SNOW    0.24398    0.01764   13.83   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Approximate significance of smooth terms:
##             edf Ref.df      F p-value    
## s(DAY)  4.62176     18  0.943 0.00109 ** 
## s(TMIN) 3.92234      4 34.384 < 2e-16 ***
## s(YEAR) 0.01468      4  0.000 0.77947    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## R-sq.(adj) =  0.219   Deviance explained = 22.2%
## -ML =  13886  Scale est. = 303.85    n = 3243
```

Choosing K?

Silk (big k) vs. Carpet (small k, min. ~3)



Other Responses, REs

- Families: gaussian, binomial, poisson, **negbin**
- Add simple (not nested) random effects to formula: `s(variable, bs = 're')`
- More REs in a GAM: `gamm4::gamm4()` (beyond our class/demo next week)

Model Assessment

- Conditions (for family used) must be met: ~~L~~INE *or* ~~L~~I, mean-variance
- All must be checked as for `(g)lm(mTMB)()` - except linearity!

Additional Checks for GAM

```
par(mar=c(4,4,2,2))
gam.check(snow.gam)
```

```
##
## Method: ML   Optimizer: outer newton
## full convergence after 7 iterations.
## Gradient range [-0.006501825,0.000858677]
## (score 13886.46 & scale 303.8515).
## Hessian positive definite, eigenvalue range [0.006419922,1621.505].
## Model rank =  28 / 28
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##              k'    edf k-index p-value
## s(DAY)  18.0000  4.6218    1.03    0.94
## s(TMIN)  4.0000  3.9223    1.04    0.99
## s(YEAR)  4.0000  0.0147    1.02    0.85
```

Prediction Plots (as usual)

```
spred <- ggpredict(snow.gam,
                   terms = c('DAY', 'YEAR[2022]'))
```

```
## Error in ggpredict(snow.gam, terms = c("DAY",
"YEAR[2022]")): could not find function "ggpredict"
```

```
plot(spred) |>
  gf_labs(title = '',
          y = 'Expected Snow (mm)', x = 'Day of
Year')
```

```
## Error in eval(expr, envir, enclos): object 'spred'
```

Which fixed values were used?

```
spred
```

```
## # Predicted values of SNOW
##
## DAY | Predicted |       95% CI
## ------------------------------
##   1 |      4.61 |  2.32, 6.90
##  47 |      2.96 |  0.58, 5.34
##  92 |      0.20 | -1.54, 1.93
## 138 |      0.79 | -0.93, 2.52
## 184 |      0.98 | -1.58, 3.54
## 229 |      0.89 | -1.60, 3.38
## 275 |      0.93 | -0.71, 2.57
## 366 |      4.61 |  2.32, 6.90
##
## Adjusted for:
## *      TMIN = 4.92
## * PREV.SNOW = 4.85
```

Shrinkage

- Some model selection is (or can be) done during model fitting
- What smooth is best? Or is the relationship a line? A flat line?
- Using *shrinkage* basis or including `select = TRUE` allows for this
- Our Default?

P-value selection

- Caution: **p-values are approximate!**
- Best when using ML (1st choice), REML (2nd choice).

```
anova(snow.gam)
```

Note: use `anova()` (not `Anova()`) for GAMs - **unlike** `lm()`, `glm()`, `glmmTMB()`. Can also use `AIC()` or `BIC()` too.

```r
anova(snow.gam)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## SNOW ~ s(DAY, k = 20, bs = "cc") + s(TMIN, k = 5, bs = "cs") +
##     s(YEAR, k = 5, bs = "cs") + PREV.SNOW
##
## Parametric Terms:
##           df     F p-value
## PREV.SNOW  1 191.3  <2e-16
##
## Approximate significance of smooth terms:
##              edf   Ref.df      F p-value
## s(DAY)   4.62176 18.00000  0.943 0.00109
## s(TMIN)  3.92234  4.00000 34.384 < 2e-16
## s(YEAR)  0.01468  4.00000  0.000 0.77947
```